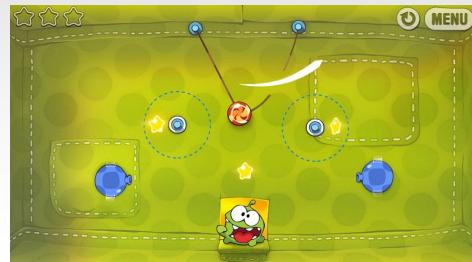


CPSC 436D

Video Game Programming



Transformations

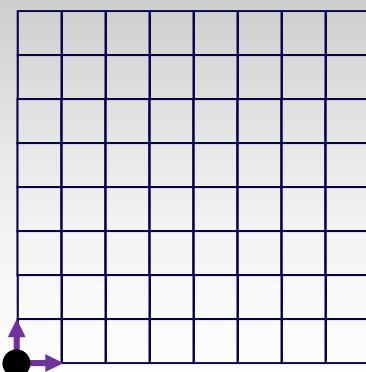


© Alla Sheffer

COORDINATE SYSTEMS



Coordinate system = Origin + Basis Vectors

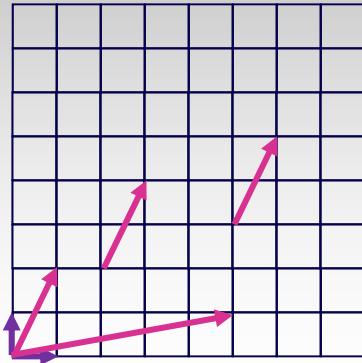


© Alla Sheffer



COORDINATE SYSTEMS

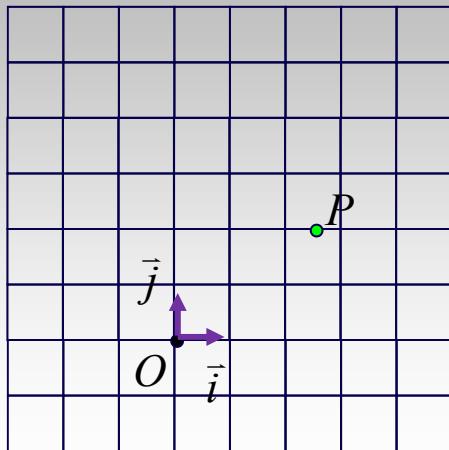
Coordinate system = Origin + Basis Vectors



© Alla Sheffer



COORDINATE SYSTEMS



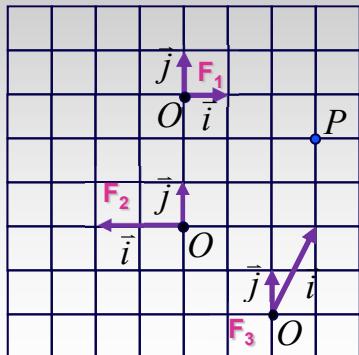
$$P = O + x\vec{i} + y\vec{j}$$

equivalent: $P = (x, y)$

© Alla Sheffer



COORDINATE SYSTEMS



$$P = O + x\vec{i} + y\vec{j}$$

\vec{F}_1

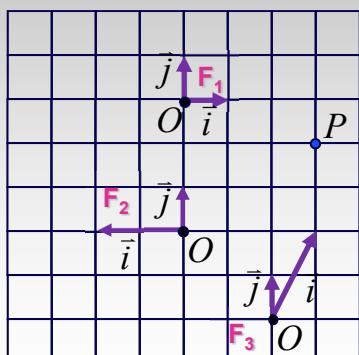
\vec{F}_2

\vec{F}_3

© Alla Sheffer



COORDINATE SYSTEMS



$$P = O + x\vec{i} + y\vec{j}$$

\vec{F}_1 P(3,-1)

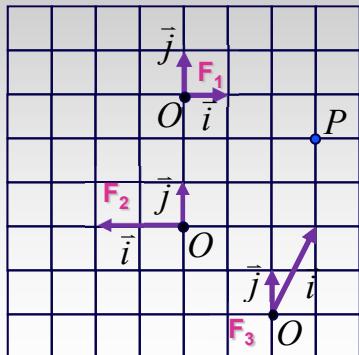
\vec{F}_2

\vec{F}_3

© Alla Sheffer



COORDINATE SYSTEMS



$$P = O + x\vec{i} + y\vec{j}$$

$$\mathbf{F}_1 \quad \mathbf{P}(3,-1)$$

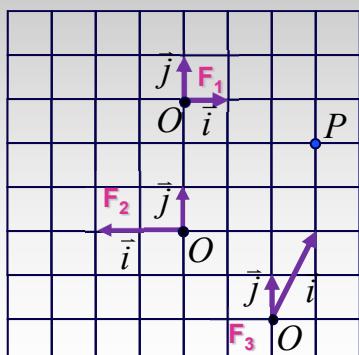
$$\mathbf{F}_2 \quad \mathbf{P}(-1.5,2)$$

$$\mathbf{F}_3$$

© Alla Sheffer



COORDINATE SYSTEMS



$$P = O + x\vec{i} + y\vec{j}$$

$$\mathbf{F}_1 \quad \mathbf{P}(3,-1)$$

$$\mathbf{F}_2 \quad \mathbf{P}(-1.5,2)$$

$$\mathbf{F}_3 \quad \mathbf{P}(1,2)$$

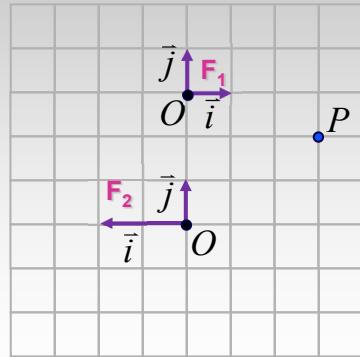
© Alla Sheffer



Transformations

Transformations as a change of frame

$$P = O + xi\vec{i} + yj\vec{j}$$



check: $P_1(3, -1)$ becomes $P_2(-1.5, 2)$

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}_1 + x_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix}_1 + y_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix}_1$$

$$\begin{bmatrix} x \\ y \end{bmatrix}_2 = \begin{bmatrix} 0 \\ 3 \end{bmatrix}_2 + x_1 \begin{bmatrix} -0.5 \\ 0 \end{bmatrix}_2 + y_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix}_2$$

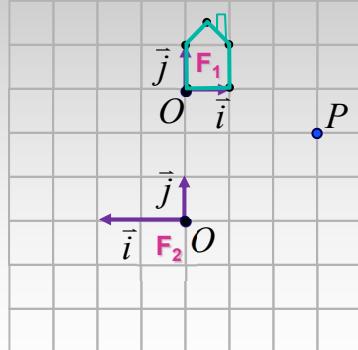
$$\begin{bmatrix} x \\ y \end{bmatrix}_2 = \begin{bmatrix} -0.5 & 0 & 0 \\ 0 & 1 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_1$$

$$P_2 = MP_1$$

© Alla Sheffer

TRANSFORMATIONS

change of basis expressed using a matrix



$$\begin{bmatrix} x \\ y \end{bmatrix}_2 = \begin{bmatrix} -0.5 & 0 & 0 \\ 0 & 1 & 3 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_1$$

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_2 = \begin{bmatrix} -0.5 & 0 & 0 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_1$$



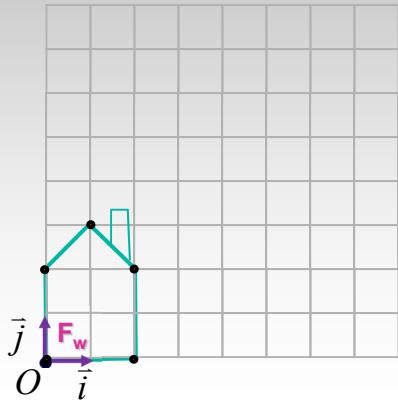
© Alla Sheffer



Usage of Transformations

set up the modeling matrix M

for each vertex v
 $v' = Mv$



© Alla Sheffer



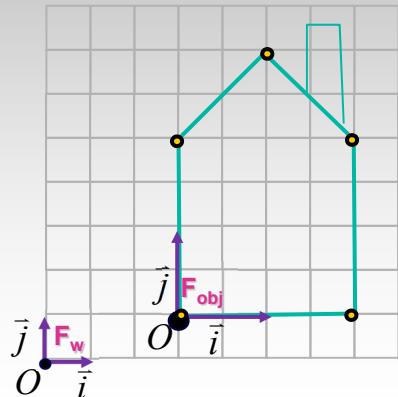
Usage of Transformations

$$P = O + x\vec{i} + y\vec{j}$$

$$\begin{bmatrix} x_{obj} \\ y_{obj} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}_{obj} + x_{obj} \begin{bmatrix} 1 \\ 0 \end{bmatrix}_{obj} + y_{obj} \begin{bmatrix} 0 \\ 1 \end{bmatrix}_{obj}$$

$$\begin{bmatrix} x \\ y \end{bmatrix}_w = \begin{bmatrix} 3 \\ 1 \end{bmatrix}_w + x_{obj} \begin{bmatrix} 2 \\ 0 \end{bmatrix}_w + y_{obj} \begin{bmatrix} 0 \\ 2 \end{bmatrix}_w$$

$$\begin{bmatrix} x \\ y \end{bmatrix}_w = \begin{bmatrix} 2 & 0 & 3 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{obj}$$



© Alla Sheffer



Using Transformations

$$\begin{bmatrix} x \\ y \end{bmatrix}_2 = \begin{bmatrix} 2 & 0 & 3 \\ 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_1$$

2D

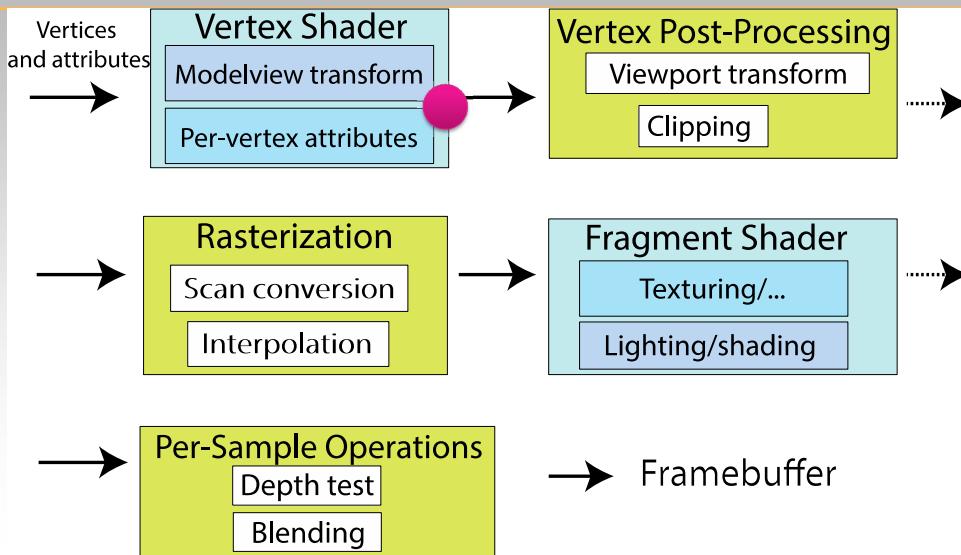
$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_w = \begin{bmatrix} 2 & 0 & 3 \\ 0 & 2 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{obj}$$

3D

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_w = \begin{bmatrix} 2 & 0 & 0 & 3 \\ 0 & 2 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_{obj}$$

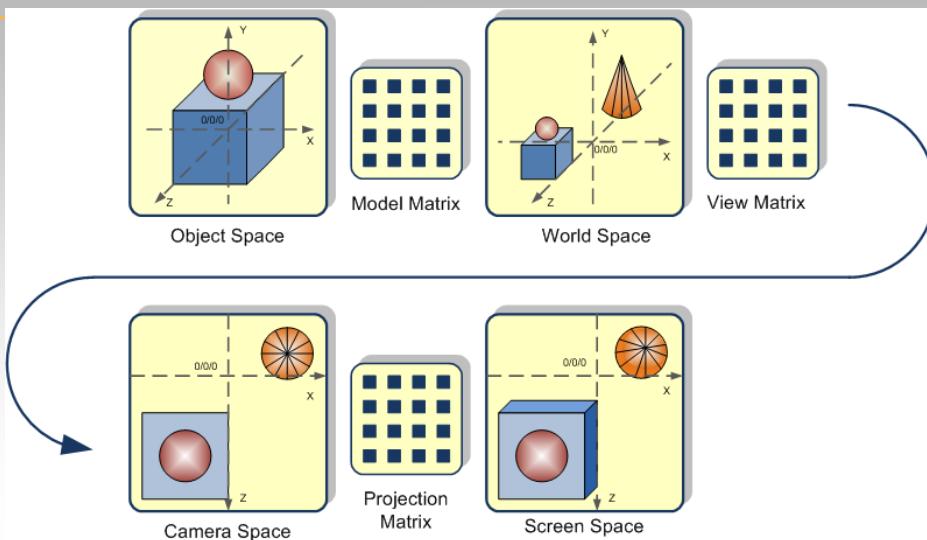
© Alla Sheffer

PIPELINE



© Alla Sheffer

coordinate systems



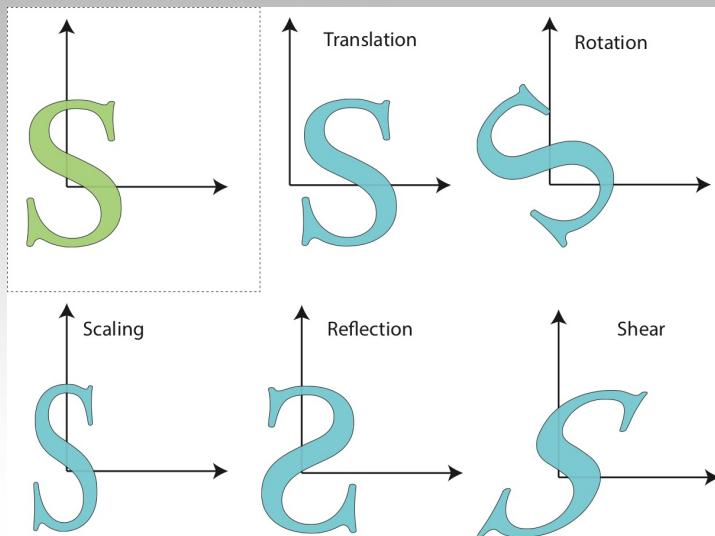
© Alla Sheffer

HOW TO TRANSFORM COORDINATES

- Between coordinate frames
- Or animate objects

© Alla Sheffer

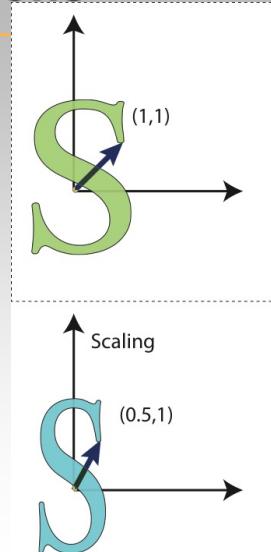
linear transformations



© Alla Sheffer

Matrix representations

Scale:



© Alla Sheffer

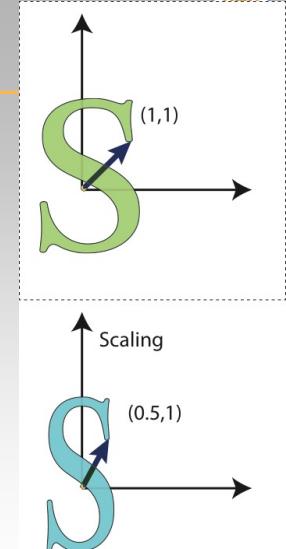
Matrix representations

Scale:

$$M = \begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix}$$

Example:

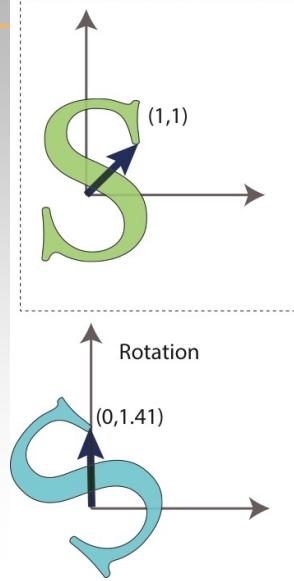
$$\begin{bmatrix} \alpha & 0 \\ 0 & \beta \end{bmatrix} \begin{pmatrix} 1 \\ 2 \end{pmatrix} = \begin{pmatrix} \alpha \\ 2\beta \end{pmatrix}$$



© Alla Sheffer

Matrix representations

Rotation



© Alla Sheffer

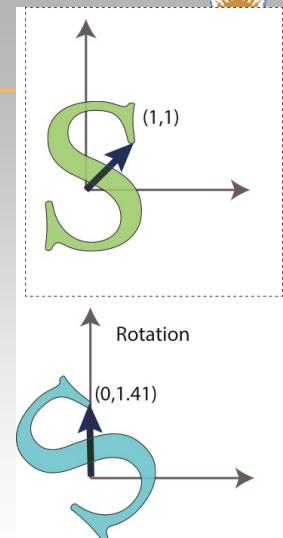
Matrix representations

Rotation

$$R(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}$$

Example:

$$\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\alpha) - \sin(\alpha) \\ \cos(\alpha) + \sin(\alpha) \end{pmatrix}$$



© Alla Sheffer

What does this 2D transformation do?

- A. Rotates by 90 deg
- B. Scales by a factor of 2
- C. Rotates by -90 deg
- D. Nothing

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

What does this 2D transformation do?

- A. Rotates by 90 deg
- B. Reflects the object
- C. Rotates by -90 deg
- D. Scales the object

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$

© Alla Sheffer



TRANSLATION

There's a minor glitch.

- Translation: can't be represented as 2x2 matrix multiplication

© Alla Sheffer



general transformations

***We need to represent all the
linear transformations + translation.***

Ideas?

$$T(\mathbf{v}) = M\mathbf{v} + \mathbf{b}$$

© Alla Sheffer



Unifying

- We want something more universal:

$$T(v) = Mv$$

- Then we could easily combine different transformations or invert
- Let's extend the vector to have one artificial coordinate: (x,y,1)
- Then what should the matrix be?

© Alla Sheffer



AUGMENTED MATRIX

$$M_{2 \times 2} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x' \\ y' \end{bmatrix}$$

$$\begin{bmatrix} M_{2 \times 2} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix}$$

Haven't changed much, have we?

© Alla Sheffer



AUGMENTED MATRIX

$$\begin{bmatrix} M_{2x2} & b_x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x' + b_x \\ y' + b_y \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} + \mathbf{b}$$

Translation

© Alla Sheffer



Affine transformations

- Linear (rotation, scaling, shear, reflections) + TRANSLATION

© Alla Sheffer



Affine transformations

- Linear (rotation, scaling, shear, reflections) + TRANSLATION
- How to convert a linear transformation matrix into affine matrix?

© Alla Sheffer



AFFINE Transformations

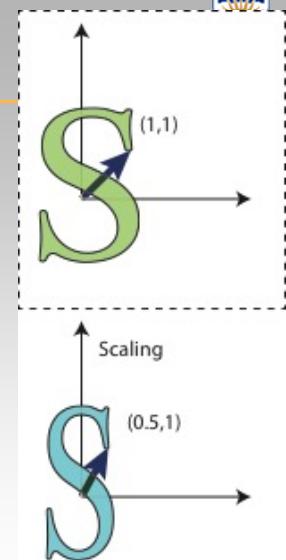
Scale:

$$M = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}$$

$$M = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Example:

$$\begin{pmatrix} a \cdot 1 \\ b \cdot 2 \\ 1 \end{pmatrix} = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$



© Alla Sheffer

AFFINE Transformations

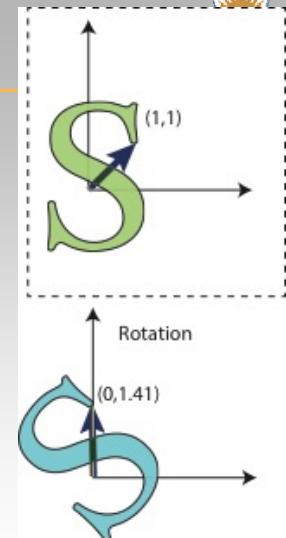
Rotation

$$M = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}$$

$$M = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Example:

$$\begin{pmatrix} a \cdot 1 \\ b \cdot 2 \\ 1 \end{pmatrix} = \begin{bmatrix} a & 0 & 0 \\ 0 & b & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$



© Alla Sheffer

AFFINE Transformations

Translation

$$M = \begin{bmatrix} 1 & 0 & C_x \\ 0 & 1 & C_y \\ 0 & 0 & 1 \end{bmatrix}$$

Example:

$$\begin{bmatrix} 1 & 0 & C_x \\ 0 & 1 & C_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} x + C_x \\ y + C_y \\ 1 \end{pmatrix}$$

© Alla Sheffer